UNITED STATES PATENT AND TRADEMARK OFFICE

UNITED STATES DEPARTMENT OF COMMERCE
United States Patent and Trademark Office
Address: COMMISSIONER FOR PATENTS
P.O. Box 1450
Alexandria, Virginia 22313-1450
www.uspto.gov

APPLICATION NO.	FILING DATE	FIRST NAMED INVENTOR	ATTORNEY DOCKET NO.	CONFIRMATION NO.
10/010,371	11/05/2001	Namik Hrle	DE92000035US1/2264P	4844

7590 11/01/2005

SAWYER LAW GROUP
P.O. Box 51418
Palo Alto, CA 94303

EXAMINER	
ORTIZ, BELIX M	
ART UNIT	**PAPER NUMBER**
2164	

DATE MAILED: 11/01/2005

Please find below and/or attached an Office communication concerning this application or proceeding.

Office Action Summary	Application No.	Applicant(s)	
	10/010,371	HRLE ET AL.	
	Examiner	Art Unit	
	Belix M. Ortiz	2164	

-- The MAILING DATE of this communication appears on the cover sheet with the correspondence address --

Period for Reply

A SHORTENED STATUTORY PERIOD FOR REPLY IS SET TO EXPIRE 3 MONTH(S) OR THIRTY (30) DAYS, WHICHEVER IS LONGER, FROM THE MAILING DATE OF THIS COMMUNICATION.

- Extensions of time may be available under the provisions of 37 CFR 1.136(a). In no event, however, may a reply be timely filed after SIX (6) MONTHS from the mailing date of this communication.
- If NO period for reply is specified above, the maximum statutory period will apply and will expire SIX (6) MONTHS from the mailing date of this communication.
- Failure to reply within the set or extended period for reply will, by statute, cause the application to become ABANDONED (35 U.S.C. § 133). Any reply received by the Office later than three months after the mailing date of this communication, even if timely filed, may reduce any earned patent term adjustment. See 37 CFR 1.704(b).

Status

1)☒ Responsive to communication(s) filed on *22 August 2005*.

2a)☒ This action is **FINAL**. 2b)☐ This action is non-final.

3)☐ Since this application is in condition for allowance except for formal matters, prosecution as to the merits is closed in accordance with the practice under *Ex parte Quayle*, 1935 C.D. 11, 453 O.G. 213.

Disposition of Claims

4)☒ Claim(s) *1,4-9,12-17 and 20-27* is/are pending in the application.

4a) Of the above claim(s) _____ is/are withdrawn from consideration.

5)☐ Claim(s) _____ is/are allowed.

6)☒ Claim(s) *1,4-9,12-17 and 20-27* is/are rejected.

7)☐ Claim(s) _____ is/are objected to.

8)☐ Claim(s) _____ are subject to restriction and/or election requirement.

Application Papers

9)☐ The specification is objected to by the Examiner.

10)☐ The drawing(s) filed on _____ is/are: a)☐ accepted or b)☐ objected to by the Examiner.

Applicant may not request that any objection to the drawing(s) be held in abeyance. See 37 CFR 1.85(a).

Replacement drawing sheet(s) including the correction is required if the drawing(s) is objected to. See 37 CFR 1.121(d).

11)☐ The oath or declaration is objected to by the Examiner. Note the attached Office Action or form PTO-152.

Priority under 35 U.S.C. § 119

12)☒ Acknowledgment is made of a claim for foreign priority under 35 U.S.C. § 119(a)-(d) or (f).

a)☒ All b)☐ Some * c)☐ None of:

1.☒ Certified copies of the priority documents have been received.

2.☐ Certified copies of the priority documents have been received in Application No. _____.

3.☐ Copies of the certified copies of the priority documents have been received in this National Stage application from the International Bureau (PCT Rule 17.2(a)).

* See the attached detailed Office action for a list of the certified copies not received.

Attachment(s)

1) ☒ Notice of References Cited (PTO-892)
2) ☐ Notice of Draftsperson's Patent Drawing Review (PTO-948)
3) ☐ Information Disclosure Statement(s) (PTO-1449 or PTO/SB/08) Paper No(s)/Mail Date _____.
4) ☐ Interview Summary (PTO-413) Paper No(s)/Mail Date. _____.
5) ☐ Notice of Informal Patent Application (PTO-152)
6) ☐ Other: _____.

DETAILED ACTION

Remarks

1. In response to communications files on 22-August-2005, claims 1, 9, and 17 are amended per applicant's request. Therefore, claims 1, 4-9, 12-17, and 20-27 are presently pending in the application.

Claim Rejections - 35 USC § 103

2. The following is a quotation of 35 U.S.C. 103(a) which forms the basis for all obviousness rejections set forth in this Office action:

> (a) A patent may not be obtained though the invention is not identically disclosed or described as set forth in section 102 of this title, if the differences between the subject matter sought to be patented and the prior art are such that the subject matter as a whole would have been obvious at the time the invention was made to a person having ordinary skill in the art to which said subject matter pertains. Patentability shall not be negatived by the manner in which the invention was made.

3. Claims 1, 4-9, 12-17, and 20-27 are rejected under 35 U.S.C. 103(a) as being unpatentable over Ponnekanti (U.S. patent 6,606,626) in view of Klein et al. (U.S. publication 2002/0038313).

As to claim 1, Ponnekanti teaches a method for reducing lock contention of concurrent transactions on a plurality of rows of a table in a relational data base system in response to a database query having a set of predicates (see column 2, lines 30-32; column 3, lines 1-9; column 3, lines 26-28; and column 20, lines 8-13), the method comprising the steps of:

(a) scanning all rows of the table within an access range determined by the query (see column 9, lines 59-62; column 9, lines 66-67; and column 10, lines 1-2), wherein the scanning step (a) further comprising the step of:

(a1) accessing the rows of the table with uncommitted read semantics, wherein the accessing is performed through any current locks on the row (see abstract; column 12, lines 46-49; and column 16, lines 53-56);

(b) evaluating each scanned row to determine whether the row satisfies the set of predicates (see column 10, lines 1-4), wherein the step of evaluating (b) include

(b1) determining that a particular row does not satisfy the set of predicates of the query (see column 3, lines 2-7 and column 3, lines 53-65).

Ponnekanti does not teach (b2) skipping the particular row, including skipping the particular row when a lock is currently held on the particular row and an update on the particular row has not yet committed while the lock is held, and continuing the scan.

Klein et al. teaches system and method for performing database operations on a continuous stream of tuples (see abstract), in which he teaches (b2) skipping the particular row, including skipping the particular row when a lock is currently held on the particular row and an update on the particular row has not yet committed while the lock is held, and continuing the scan (see paragraph 18).

It would have been obvious to a person having ordinary skill in the art at the time the invention was made to have modified Ponnekanti by the teaching of Klein et

al., because (b2) skipping the particular row, including skipping the particular row when a lock is currently held on the particular row and an update on the particular row has not yet committed while the lock is held, and continuing the scan, would enable the method because "Generally, the lock in the conflicting mode will be held by a transaction other than the transaction associated with the operator in question. When the SQL statement being executed uses the new "skip conflict" syntax provided by the present invention, the operator skips over rows that are locked in a conflicting mode, which would otherwise cause the operator to suspend operation. Furthermore, if the operator is operating in streaming mode, a key that identifies such skipped rows is added to the operator's list of rows to be processed during the delta scan phase", (see Klein et al., paragraph 18).

As to claim 4, Ponnekanti as modified teaches wherein the returning step (b3) further comprises the steps of:

requesting a lock on the row that satisfies the set of predicates (see Ponnekanti, column 3, lines 46-50 and column 3, lines 62-63);

suspending the scan, if the requested lock is refused (see Ponnekanti, column 4, lines 10-11);

repeating the request for a lock and re-evaluating the row when the lock is permitted (see Ponnekanti, column 12, lines 52-54 and column 12, lines 61-67); and

returning the row if the row still satisfies the set of predicates of the query

(see Ponnekanti, column 3, lines 62-63).

As to claim 5, Ponnekanti as modified teaches wherein the returning step (b3) further comprises the step of:

releasing the lock, skipping the row, and continuing the scan if the row no longer satisfies the set of predicates of the query (see Ponnekanti, column 16, lines 42-62).

As to claim 6, Ponnekanti as modified teaches wherein the returning step (b3) further includes the step of:

returning the row as a result set (see Ponnekanti, column 3, lines 62-63).

As to claim 7, Ponnekanti as modified teaches wherein the returning step (b3) further includes the step of:

returning the row if the row is a committed row (see Ponnekanti, column 15, lines 8-10).

As to claim 8, Ponnekanti as modified teaches wherein the database query is a SQL statement (see Ponnekanti, column 1, lines 65-67).

As to claim 9, Ponnekanti teaches an apparatus for reducing lock contention of concurrent transactions on a plurality of rows of a table in a

relational data base system in response to a database query having a set of predicates (see figure 1A; column 2, lines 30-32; column 3, lines 1-9; column 3, lines 26-28; and column 20, lines 8- 3), comprising:

means for scanning all rows of the table within an access range determined by the query (see column 9, lines 59-62; column 9, lines 66-67; and column 10, lines 1-2), wherein means for the scanning further comprising:

means for accessing the rows of the table with uncommitted read semantics, wherein the accessing is performed through any current locks on the rows (see column 12, lines 46-49 and column 16, lines 53-56);

means for evaluating each scanned row to determine whether the row satisfies the set of predicates (see column 10, lines 1-4), wherein the means for evaluating include:

means for determining that a particular row does not satisfy the set of predicates of the query (see column 3, lines 64-65); and

Ponnekanti does not teach means for skipping the particular row, including skipping the particular row when a lock is currently held on the particular row and an update on the particular row has not yet committed while the lock is held, and continuing the scan.

Klein et al. teaches system and method for performing database operations on a continuous stream of tuples (see abstract), in which he teaches means for skipping the particular row, including skipping the particular row when a lock is

currently held on the particular row and an update on the particular row has not yet committed while the lock is held, and continuing the scan (see paragraph 18).

It would have been obvious to a person having ordinary skill in the art at the time the invention was made to have modified Ponnekanti by the teaching of Klein et al., because means for skipping the particular row, including skipping the particular row when a lock is currently held on the particular row and an update on the particular row has not yet committed while the lock is held, and continuing the scan, would enable the method because "Generally, the lock in the conflicting mode will be held by a transaction other than the transaction associated with the operator in question. When the SQL statement being executed uses the new "skip conflict" syntax provided by the present invention, the operator skips over rows that are locked in a conflicting mode, which would otherwise cause the operator to suspend operation. Furthermore, if the operator is operating in streaming mode, a key that identifies such skipped rows is added to the operator's list of rows to be processed during the delta scan phase", (see Klein et al., paragraph 18).

As to claim 12, Ponnekanti teaches wherein the means for returning step further comprising:

means for requesting a lock on the row (see Ponnekanti, column 3, lines 46-50);

means for suspending the scan, if the requested lock is refused (see Ponnekanti, column 4, lines 10-11);

means for repeating the request for a lock and re-evaluating the row when the lock is permitted (see Ponnekanti, column 12, lines 52-54 and column 12, lines 61-67); and

means for returning the row if the row still satisfies the set of predicates of the query (see Ponnekanti, column 3, lines 62-63).

As to claim 13, Ponnekanti as modified teaches wherein the means for returning step further includes means for releasing the lock, skipping the row, and continuing the scan if the row no longer satisfies the set of predicates of the query (see Ponnekanti, column 16, lines 42-62).

As to claim 14, Ponnekanti as modified teaches wherein the returned row is returned as a result set (see Ponnekanti, column 3, lines 62-63).

As to claim 15, Ponnekanti as modified teaches wherein the row returned is a committed row (see Ponnekanti, column 15, lines 8-10).

As to claim 16, Ponnekanti as modified teaches wherein the database query is a SQL statement (see Ponnekanti, column 1, lines 65-67).

As to claim 17, Ponnekanti teaches a computer readable medium containing programming instructions for reducing lock contention of concurrent

transactions on a plurality of rows of a table in a relational data base system in response to a database query having a set of predicates (see column 2, lines 30-32; column 3, lines 1-9; column 3, lines 26-28; column 6, lines 66-67; column 7, lines 1-9; and column 20, lines 8-13), the programming instructions for:

(a) scanning all rows of the table within an access range determined by the query (see column 9, lines 59-62; column 9, lines 66-67; and column 10, lines 1-2), wherein the scanning instruction (a) further comprising the instruction for:

(a1) accessing the rows of the table with uncommitted read semantics, wherein the accessing is performed through any current locks on the rows (see column 12, lines 46-49 and column 16, lines 53-56);

(b) evaluating each scanned row to determine whether the row satisfies the set of predicates (see column 10, lines 1-4), wherein the instruction for evaluating (b) further comprises the instruction for:

(b1) determining that a particular row does not satisfy the set of predicates of the query (see column 3, lines 2-7 and column 3, lines 53-65); and

Ponnekanti does not teach (b2) skipping the particular row, including skipping the particular row when a lock is currently held on the particular row and an update on the particular row has not yet committed while the lock is held, and continuing the scan.

Klein et al. teaches system and method for performing database operations on a continuous stream of tuples (see abstract), in which he teaches (b2) skipping the particular row, including skipping the particular row when a lock is currently held

on the particular row and an update on the particular row has not yet committed while the lock is held, and continuing the scan (see paragraph 18).

It would have been obvious to a person having ordinary skill in the art at the time the invention was made to have modified Ponnekanti by the teaching of Klein et al., because (b2) skipping the particular row, including skipping the particular row when a lock is currently held on the particular row and an update on the particular row has not yet committed while the lock is held, and continuing the scan, would enable the method because "Generally, the lock in the conflicting mode will be held by a transaction other than the transaction associated with the operator in question. When the SQL statement being executed uses the new "skip conflict" syntax provided by the present invention, the operator skips over rows that are locked in a conflicting mode, which would otherwise cause the operator to suspend operation. Furthermore, if the operator is operating in streaming mode, a key that identifies such skipped rows is added to the operator's list of rows to be processed during the delta scan phase", (see Klein et al., paragraph 18).

As to claim 20, Ponnekanti as modified teaches wherein the returning step instruction (b3) further comprises the instruction for:

requesting a lock on the row (see Ponnekanti, column 3, lines 46-50);

suspending the scan, if the requested lock is refused (see Ponnekanti, column 4, lines 10-11);

repeating the request for a lock and re-evaluating the row when the

lock is permitted (see Ponnekanti, column 12, lines 52-54 and column 12, lines 61-67); and

returning the row if the row still satisfies the set of predicates of the query (see Ponnekanti, column 3, lines 62-63).

As to claim 21, Ponnekanti as modified teaches wherein the returning instruction (b3) further comprises the instruction for:

releasing the lock, skipping the row, and continuing the scan if the row no longer satisfies the set of predicates of the query (see Ponnekanti, column 16, lines 42-62).

As to claim 22, Ponnekanti as modified teaches wherein the returning instruction (b3) further includes the instruction for:

returning the row as a result set (see Ponnekanti, column 3, lines 62-63).

As to claim 23, Ponnekanti as modified teaches wherein the returning instruction (b3) further includes the instruction for:

returning the row if the row is a committed row (see Ponnekanti, column 15, lines 8-10).

As to claim 24, Ponnekanti as modified teaches wherein the database query is a SQL statement (see Ponnekanti, column 1, lines 65-67).

As to claim 25, Ponnekanti as modified teaches wherein the step of evaluating (b) includes

(b3) determining that a row satisfies the set of predicates of the query, and returning the row (see Ponnekanti, column 3, lines 62-63 and column 4, lines 21-22).

As to claim 26, Ponnekanti as modified teaches wherein the means for evaluating includes means for determining that a row satisfies the set of predicates of the query, and for returning the row (see Ponnekanti, column 3, lines 62-63 and column 4, lines 21-22).

As to claim 27, Ponnekanti as modified teaches wherein the instruction of evaluating (b) includes instruction for:

(b3) determining that a row satisfies the set of predicates of the query, and returning the row (see Ponnekanti, column 3, lines 62-63 and column 4, lines 21-22).

Conclusion

Applicant's amendment necessitated the new ground(s) of rejection presented in this Office action. Accordingly, **THIS ACTION IS MADE FINAL**. See MPEP § 706.07(a). Applicant is reminded of the extension of time policy as set forth in 37 CFR 1.136(a).

A shortened statutory period for reply to this final action is set to expire THREE MONTHS from the mailing date of this action. In the event a first reply is filed within TWO MONTHS of the mailing date of this final action and the advisory action is not mailed until after the end of the THREE-MONTH shortened statutory period, then the shortened statutory period will expire on the date the advisory action is mailed, and any extension fee pursuant to 37 CFR 1.136(a) will be calculated from the mailing date of the advisory action. In no event, however, will the statutory period for reply expire later than SIX MONTHS from the date of this final action.

Any inquiry concerning this communication or earlier communications from the examiner should be directed to Belix M. Ortiz whose telephone number is 571-272-4081. The examiner can normally be reached on moday-friday 9am-5pm.

The fax phone number for the organization where this application or proceeding is assigned is 571-273-8300.

Information regarding the status of an application may be obtained from the Patent Application Information Retrieval (PAIR) system. Status information for published applications may be obtained from either Private PAIR or Public PAIR. Status information for unpublished applications is available through Private PAIR only. For more information about the PAIR system, see http://pair-direct.uspto.gov. Should you have questions on access to the Private PAIR system, contact the Electronic Business Center (EBC) at 866-217-9197 (toll-free).

C. Rones

bmo

CHARLES RONES
SUPERVISORY PATENT EXAMINER

October 25, 2005.